Statistics for Data Science | Exercise 14
Wintersemester 2023/24
Please complete these problems before the exercise session on Tuesday 6 February, 2024, 8:30. Please be prepared to present your solutions to any problems that you completed successfully.

1. Consider a bivariate Gaussian distribution

$$\begin{pmatrix} x_1 \\ x_2 \end{pmatrix} \sim \mathcal{N}\left(\begin{pmatrix} 0 \\ 0 \end{pmatrix}, \begin{pmatrix} 1 & p \\ p & 1 \end{pmatrix}\right).$$

   (a) Write a Gibbs sampler for this distribution. Based on the generated sample, what are the expected value of $(x_1, x_2)^{\mathrm{T}}$ and the marginal standard deviations of x_1 and x_2?

   (b) Repeat part (a) for parameter values $p = 0.5, 0.9, 0.99$, and 0.999. How does the degree of correlation between x_1 and x_2 affect the performance of the Gibbs sampler?

2. Consider the mathematical model

$$y = \begin{pmatrix} x_1^2 + x_2^2 \\ x_2 \end{pmatrix} + \eta,$$

   where $y \in \mathbb{R}^2$ is the measurement and $x = (x_1, x_2)^{\mathrm{T}} \in \mathbb{R}^2$ is the unknown. Let us set the prior $x = z \cdot \mathbf{1}_{[-4,4]^2}(z)$, where $z \sim \mathcal{N}((0,0)^{\mathrm{T}}, I)$,

$$\mathbf{1}_B(z) = \begin{cases} 1, & z \in B, \\ 0, & \text{otherwise,} \end{cases}$$

   and $\eta \sim \mathcal{N}(0, \delta^2 I)$ with $\delta = 0.1$. Suppose we are given the observation $y = (7, -2)^{\mathrm{T}}$. Implement MCMC with the Metropolis–Hastings kernel

$$x_{k+1} \sim \sqrt{1-\beta^2} \cdot x_k + \beta\xi, \quad \xi \sim \mathcal{N}(0, I),$$

   for different values of $\beta \in (0,1)$ to sample the posterior density. For each value of β produce 10 000 samples and plot them. What do you notice? Also compute for each β the *acceptance ratio*, i.e., the ratio between accepted jumps and the total length of the chain. Use the origin as initial value.

   Using the best choice of β, compute the posterior mean, i.e., the conditional mean estimator

$$\hat{x}_{\mathrm{CM}} = \int_{\mathbb{R}^2} x f(x|y)\,\mathrm{d}x.$$

The exercises continue on the next page!

3–4. (This task is worth 2 points.) The random walk Metropolis algorithm scales poorly with increasing dimension.† Meanwhile, the single component Gibbs sampler is computationally expensive for high-dimensional problems. A surprisingly effective alternative is the so-called *Metropolis-within-Gibbs algorithm*, which combines the powerful Gibbs sampler with the computationally inexpensive Metropolis algorithm. The algorithm to draw a sample from the d-dimensional probability density function f can be described as follows:

1. Choose the initial value $x^{(0)} \in \mathbb{R}^d$ and set $k = 0$.
2. Draw the next sample as follows:
   (i) Set $x = x^{(k)}$ and $j = 1$.
   (ii) Draw $t \in \mathbb{R}$ from the one-dimensional distribution
   $$f(t|y_1, \ldots, y_{j-1}, x_{j+1}, \ldots, x_d) \propto f(y_1, \ldots, y_{j-1}, t, x_{j+1}, \ldots, x_d)$$
   by performing one step of the Metropolis algorithm and set $y_j = t$.
   (iii) If $j = d$, set $y = (y_1, \ldots, y_d)$ and terminate the inner loop. Otherwise, set $j \leftarrow j + 1$ and return to step (ii).
3. Set $x^{(k+1)} = y$, increase $k \leftarrow k + 1$ and return to step 2.

Suppose that we are interested in estimating a signal $g\colon [0, 1] \to \mathbb{R}$ from noisy, blurred observations modeled by

$$y_i = y(s_i) = \int_0^1 K(s_i, t)g(t)\,\mathrm{d}t + \varepsilon_i, \quad i \in \{1, \ldots, k\}, \tag{1}$$

where $s_i = \frac{i}{k} - \frac{1}{2k}$ for $i \in \{1, \ldots, k\}$, the blurring kernel is

$$K(s, t) = \exp\left(-\frac{1}{2 \cdot 0.05^2}(s - t)^2 \right),$$

and we have i.i.d. Gaussian measurement noise $\varepsilon_i \sim \mathcal{N}(0, \sigma^2)$ with $\sigma = 10^{-3}$. As we discussed during last week's lecture, the integral equation (1) can be discretized using the midpoint rule with points $t_j = \frac{j}{d} - \frac{1}{2d}$, $j \in \{1, \ldots, d\}$, to obtain the linear measurement model

$$y = Ax + \varepsilon, \tag{2}$$

where $y = [y_1, \ldots, y_k]^{\mathrm{T}} \in \mathbb{R}^k$ is the measurement, $A = \left(\frac{1}{d}K(s_i, t_j)\right)_{\substack{i=1,\ldots,k \\ j=1,\ldots,d}} \in \mathbb{R}^{k \times d}$ is the system matrix, and $x = [g(t_1), \ldots, g(t_d)]^{\mathrm{T}} \in \mathbb{R}^d$ is the unknown.

Download the file `signal.mat` from the course website. The file contains the objects `y`, `A`, and `t` corresponding to the noisy, blurred signal y, the system matrix A, and the vector t, respectively. The file can be imported in Python with the command

The exercises continue on the next page!

†The preconditioned Crank–Nicolson (pCN) method considered in task 2 can be used to carry out dimension-robust sampling, but it requires careful tuning of the free parameter $\beta \in (0, 1)$.

```
data = scipy.io.loadmat('signal.mat')
```

and you can access the objects by calling `data['y']`, `data['A']`, and `data['t']`. Note that $k = d = 100$.

Suppose that we know *a priori* that the true signal x corresponds to a piecewise constant function $g\colon [0,1] \to \mathbb{R}$. A reasonable choice for the prior would then be the so-called *anisotropic total variation prior*

$$f(x) \propto \exp\Bigg(-\lambda \sum_{k=1}^{d} |x_{k+1} - x_k| \Bigg), \quad \lambda > 0, \tag{3}$$

where we assume periodic boundary conditions, i.e., $x_{d+1} = x_1$.

Your task is as follows:

Write down the posterior density $f(x|y)$ for the unknown parameter $x \in \mathbb{R}^d$ in (2) using the prior (3) with $\lambda = 100$. Use the Metropolis-within-Gibbs algorithm with random walk Metropolis step size $\gamma = 0.05$ to draw a sample of size $N = 10^4$ from the posterior density, and approximate the CM estimator $\hat{x}_{\mathrm{CM}}$ of the unknown parameter x by computing the sample average. Finally, visualize the approximate CM estimator you obtained by plotting it as a function of `t`.

Hint: Your reconstruction should look a bit like the boxcar function $\mathbf{1}_{[0.3,0.7]}$, which was the function used to generate the measurement data.